Claims

What is claimed is:

1. A method of performing a search of a numerical DOM, comprising the steps of:

(a) receiving a query;

(b) when the query is a fully qualified query, transforming a target string to form a fully qualified hashing code;

(c) performing an associative lookup in a map index using the fully qualified hashing code;

(d) returning a map offset;

(e) returning a data couplet.

2. The method of claim 1, further including the steps of:

(f) converting an identified couplet of the numerical DOM into an XML string.

3. The method of claim 2, further including the steps of:

(g) converting the data couplet into a data XML string.

4. The method of claim 1, wherein step (a) further includes the steps of:

(a1) when the query is a partially qualified query, transforming a target to form a partially qualified hashing code;

(a2) performing an associative lookup in a dictionary index using the partially qualified hashing code;

(a3) returning a dictionary offset;

(a4) locating the complete string in the dictionary, using the dictionary offset;

(a5) locating a pointer in a map index using the complete string;

(a6) locating the complete reference in the numerical DOM using the pointer.

5. The method of claim 1, wherein step (a) further includes the steps of:

(a1) when the query includes a wildcard target, scanning a dictionary for the wildcard target;

(a2) returning a complete string from the dictionary that contains the wildcard target;

(a3) locating a pointer in a map index using the complete string;

(a4) locating a couplet in the numerical DOM using the pointer.

6. A method of performing a search of a numerical DOM comprising the steps of:

(a)   receiving a query;

(b)   determining a target type of the query;

(c)   when the target type is an incomplete data string, performing a sliding window search of a dictionary;

(d)   returning a complete data string; and

(e)   returning an incomplete data couplet.

7. The method of claim 6, wherein step (d) further includes the step of returning a plurality of dictionary offsets.

8. The method of claim 6, further including the steps of:

(f)   when the target type is an incomplete tag and a complete data string, transforming the incomplete tag to form an incomplete target hashing code;

(g)   performing an associative lookup in a map index using the incomplete tag hashing code;

(h)   returning at least one map offset.

9. The method of claim 8, further including the steps of:

(i)   transforming the complete data string to form a complete data string hashing code;

(j)   performing an associative lookup in the map index;

(k)   returning a data string map offset;

(l)   comparing at least one map offset and the data string map offset.

10.   The method of claim 1, wherein step (b) further includes the step of:

(b1) performing a linear feedback shift register operation on the target string to from the fully qualified hashing code.

11. The method of claim 4, wherein step (a2) further includes the step of:

(i) performing a modulo two polynomial division on the target to form the partially qualified hashing code.

12. A method of translating a structured data document, comprising the steps of:

(a) creating a numerical DOM of the structured data document;

(b) translating a first format dictionary of the numerical DOM into a second format dictionary; and

(c) adding a second set of dictionary pointers, the second set of dictionary pointers pointing to offsets in the second format dictionary.

13. The method of claim 12, further including the steps of:

(d) converting a plurality of dictionary offset pointers to a plurality of dictionary index pointers.

14. A method of creating an alias in a numerical DOM, comprising the steps of:

(a) receiving an alias request;

(b) finding a dictionary offset for the original string in a dictionary; and

(c) converting the original string to the alias at the dictionary offset.

15. The method of claim 14, further including the steps of:

(d) creating an alias index.

16. The method of claim 14, wherein step (b) further includes the steps of:

(b1) transforming the original string to form a string hashing code;

(b2) performing an associative lookup in the dictionary to find the dictionary offset.

17. The method of claim 15, wherein step (d) includes creating an array that matches the dictionary offset to the original string.

18. A method of performing a search of a numerical DOM, comprising the steps of:

(a) receiving a query;

(b) transforming the query to form a fully qualified hashing code;

(c) performing an associative lookup in a map index using the fully qualified hashing code; and

(d) returning a map offset.

19. The method of claim 18, further including the steps of:

(e) converting an identified couplet of the numerical DOM into an XML string.

20. The method of claim 18, further including the steps of:

(e) determining if the target is a complete data string;

(f) when the target is a complete data string, transforming the complete data string to form a complete hashing code;

(g) performing an associative lookup in a dictionary index using the complete data hashing code;

(h) returning a dictionary offset;

(i) scanning the numerical DOM for the dictionary offset;

(j) returning a data couplet.

21. The method of claim 20, further including the steps of:

(k) converting the data couplet into a data XML string.

22. The method of claim 18, further including the steps of:

(e) determining if the target is a wildcard data string;

(f) when the target is the wildcard data string, performing a sliding window search of a dictionary;

(g) returning a dictionary offset of a match;

(h) scanning the numerical DOM for the dictionary offset;

(i) returning an incomplete data couplet.